The following is a copy of Applicants' claims that identifies language being added with underlining ("___") and language being deleted with strikethrough ("——"), as is applicable:

1. (Currently Amended) A method of controlling a local process that forms part of a first processing entity, said first processing entity maintaining a plurality of associations with a plurality of remote processes in a second processing entity, said method comprising the steps of:

- receiving at a computer executing the local process a failure message from a remote process indicating a fault affecting an association linking the local process with that remote process, wherein the plurality of associations comprise transport connections between respective local processes and remote processes;

- queuing, at the computer executing the local process, data messages destined for that remote process;

- controlling the transmission of an acknowledgement of the failure message at the computer executing the local process so that data messages pending on the association are received at that remote process via an alternate path before the acknowledgment of the failure message; and

- initiating a traffic diversion to set up an the alternate path between said local process of said first processing entity and said second processing entity for queued data messages by avoiding the failed association.

2. (Original) A method as claimed in claim 1 wherein the controlling comprises delaying the acknowledgment of the failure message.

3. (Original) A method as claimed in claim 2 wherein the delay is for a predeterminable time period.

4. (Original) A method as claimed in claim 2 wherein the delay is determined by transmission and acknowledgment of a heartbeat message.

5. (Original) A method as claimed in claim 1 wherein the controlling comprises sending the acknowledgement of the failure message on the data stream used for the data messages.

6. (Previously Presented) A method as claimed in claim 1 comprising testing the association to determine if the association is active and, if not, dropping messages queued for the association.

7. (Previously Presented) A method as claimed in claim 1 wherein the first processing entity maintains a plurality of associations between a plurality of local processes and a plurality of remote processes.

8. (Original) A method as claimed in claim 7 comprising informing other local processes of the fault so that such other local processes can avoid involving the failed association in traffic diversion procedures initiated by them.

9. (Original) A method as claimed in claim 7 or claim 8 comprising determining whether pending messages form part of a stateful transaction, and, if so, finding an alternative local process to provide an alternative path to the same remote process.

10. (Original) A method as claimed in claim 9 wherein the determining comprises determining whether the messages are TCAP messages.

11. (Original) A method as claimed in claim 7 or claim 8 comprising determining whether pending messages form part of a stateless transaction, and, if so, finding an alternative path through the same local association to another remote process.

12. (Original) A method as claimed in claim 11 wherein the determining comprises determining whether the messages are non TCAP messages.

13. (Previously Presented) A method as claimed in claim 1 wherein the traffic diversion comprises modifying routing tables.

14. (Previously Presented) A method as claimed in claim 1 wherein the first processing entity is a signalling gateway, the local processes being signalling gateway processes having a common point code or set of point codes.

15. (Previously Presented) A method as claimed in claim 1 wherein the second processing entity is an application server, the remote processes being application server processes having a common routing key.

16. (Original) A method as claimed in claim 15 wherein the message indicating the fault is an ASP_INACTIVE or ASP_DOWN message and the acknowledgement being respectively an ASP_INACTIVE_ACK message or an ASP_DOWN_ACK message.

17. (Previously Presented) A method as claimed in claim 1 further comprising the initiating of a switch back procedure to include a new association linking a local process with a remote process.

18. (Original) A method as claimed in claim 17 comprising informing other local processes of the new association so that such other local processes can begin involving the new association.

19. (Previously Presented) A method as claimed in claim 1 wherein the associations are SCTP associations.

20. (Currently Amended) A computer readable storage medium embedded with computer code for controlling a local process using a method as claimed in claim 1, wherein the computer code is executed by a computer to perform said method.

21. (Previously Presented) A signalling gateway comprising a plurality of local processes that are controlled using a computer program code as claimed in claim 20.

22. (Currently Amended) A method of recovering failure in a distributed signalling gateway maintaining a plurality of associations between signalling gateway processes of said distributed signalling gateway and application server processes of an application server, said method comprising the steps of:

- initiating a traffic diversion in response to a failure message to set up an alternate path between said signalling gateway processes and said application server processes in case of fault affecting an association;

- initiating a switch back to include a new association linking a signalling gateway process and an application server process, each application server process being connected to each signalling gateway process through the association, wherein the association comprises a transport connection between respective signalling gateway process and application server process;

- according to the change of status of any association, updating routing tables capable of routing data messages received by said signalling gateway processes to its destined application server processes; and

- distributing sequentially messages from said signalling gateway to said plurality of application server processes according to said routing tables.

23. (Original) The method as claimed in claim 22 wherein said step of initiating a traffic diversion further comprising the steps of:

- starting a protection timer; queuing messages destined for the application server process of the failed association;

- informing other signalling gateway processes of the fault so that other signalling gateway processes can avoid involving the failed association in traffic diversion procedure initiated by them;

- controlling the transmission of an acknowledgement of the failure message so that data messages pending on the association are received at the application server process before the acknowledgement; and

- finding alternate path to forward subsequent stateless processing messages onto another application server process through another association or to forward subsequent stateful processing messages through an alternate signalling gateway process still associated with the same application server process.

24. (Original) The method as claimed in claim 23 wherein said step of finding alternate path to forward subsequent stateless or stateful processing messages further comprising the steps of:

- re-computing said routing tables for said application server if the traffic is carrying stateless processing messages, sending messages according to said newly updated routing tables if there are still entry left in said routing tables and continuing to process until no entry is left in said routing tables; and

- finding an active signalling gateway process to divert the traffic for said application service process if the traffic is carrying stateful processing message, and sending said stateful processing messages onto said signalling gateway process through said alternate path.

25. (Original) The method as claimed in any of claims 22 to 24 wherein said step of initiating a switch back to include a new association further comprises the steps of:

- starting a protection timer further to the reception of an association activation;

- queuing data messages destined to the application server process of the new association;

- controlling the transmission of an acknowledgement of the association activation so that all diverted data messages have been transmitted via a diversion path;

- informing other signalling gateway processes of said new association; and

- re-computing said routing tables.

26. (Previously Presented) The method as claimed in claim 22 wherein said signalling gateway is coupled to a signalling end point across a signalling system No. 7 network.

27. (Previously Presented) The method as claimed in claim 22 wherein each signalling gateway process of said signalling gateway is coupled to each application server process across an internet protocol network.

28. (Previously Presented) The method as claimed in claim 23 wherein said stateful and stateless processing messages are respectively TCAP and non-TCAP messages identified by transaction identification numbers.

29. (Previously Presented) The method as claimed in claim 22 wherein said routing tables used for distributing signalling messages from said plurality of signalling gateway processes to said plurality of application server processes are SLS routing tables.